Lecture 18

Fourier-based learning algorithms

- the low degree algorithm
- Fourier Concentration
- Noise sensitivity

Review:

Recall Fourier Transform:

$$\chi_S(x) = \prod_{i\in S} x_i$$

$$\langle f,g\rangle = \frac{1}{2^n}\sum_x f(x)\,g(x)$$

$$\hat{f}(S) = \langle f,\chi_S\rangle \overset{\text{lemma}}{\swarrow\!=} 1 - 2\cdot\Pr[f(x)\neq\chi_S(x)]$$

$$\forall f,\quad f(x) = \sum \hat{f}(S)\,\chi_S(x)$$

Plancherel $\langle f,g\rangle = \sum_S \hat{f}(S)\,\hat{g}(S)$

"Boolean" Parseval $1 \underset{\substack{\uparrow\\ f(x)\in\{+1,-1\}}}{=} \langle f,f\rangle \underset{\substack{\uparrow\\ \text{Plancherel}}}{=} \sum_S \hat{f}(S)^2$

Learning via Fourier Representation

will look at learning algorithms that are based on estimating Fourier representation of fctn f (similar to polynomial interpolation)

Approximating one Fourier coefficient:

lemma for any $S \subseteq [n]$, can approx $\hat{f}(S)$ to within additive γ

(i.e. $|\text{output} - \hat{f}(S)| \le \gamma$)

with prob $\ge 1-\delta$ in $O\left(\frac{1}{\gamma^2} \log \frac{1}{\delta}\right)$ samples.

no queries needed!

(Proved last time)

Today:

The low degree algorithm

definition of fctns for which low degree Fourier coeffs pretty much suffice to describe fctn:

def $f: \{\pm 1\}^n \to \mathbb{R}$ has $\alpha(\varepsilon, n)$ - Fourier concentration

if $$\sum_{\substack{S \subseteq [n] \\ \text{s.t.} \\ |S| > \alpha(\varepsilon, n)}} \hat{f}(S)^2 \leq \varepsilon \qquad \forall \; 0 < \varepsilon < 1$$

↑ for Boolean f, this implies

$$\sum_{\substack{S \subseteq [n] \\ \text{s.t.} \\ |S| \leq \alpha(\varepsilon, n)}} \hat{f}(S)^2 \geq 1 - \varepsilon$$

examples

1) fctn f which depends on $\leq k$ vars

has $$\sum_{\substack{S \text{ s.t.} \\ |S| > k}} \hat{f}(S)^2 = 0$$

if f doesn't depend on x_i then all $\hat{f}(S)$ for which $i \in S$ satisfy $\hat{f}(S) = 0$

Low degree algorithm

approximates fctns with $d \equiv \alpha(\varepsilon, n)$ Fourier concentration:

Given: d degree
τ accuracy
δ confidence

Algorithm:

- Take $m = O\left(\frac{n^d}{\tau} \ln \frac{n^d}{\delta}\right)$ samples
- For each S s.t. $|S| \le d$:
  $C_S \leftarrow$ estimate of $\hat{f}(S)$

$\le \binom{n}{d}$ of these

reuse samples

- let $h(x) \equiv \sum_{|S| \le d} C_S \cdot \chi_S(x)$
- output sign(h) as hypothesis

Why does this work?

Two stages:

$\frac{L_2\text{-dist}}{2^n}$

1) Show that f has low F.C.

$$\Longrightarrow \quad E_x[(f(x)-h(x))^2] \text{ small}$$

2) Show that $\Pr[f(x) \neq \text{sign}(h(x))] \leq E_x[(f(x)-h(x))^2]$

↑ Hamming dist

put together: f has low F.C. $\Longrightarrow$ sign$(h(x))$ is good approximation of f

First "stage":

Thm1 if f has $d = \alpha(\varepsilon, n)$ - F.C. then
h satisfies $E_x[(f(x)-h(x))^2 \leq \varepsilon + \tau$
with prob $\geq 1-\delta$ (Proved last time)

2nd "stage":

Thm 2 $f: \{\pm1\}^n \to \{\pm1\}$
$h: \{\pm1\}^n \to \mathbb{R}$
then $\Pr[f(x) \neq \text{sign}(h(x))] \leq \underset{x\in U}{E}[(f(x)-h(x))^2]$

Proof.

$$\underset{x \in U}{E}\left[(f(x)-h(x))^2\right] = \frac{1}{2^n} \sum_x (f(x)-h(x))^2$$ defn.

$$\Pr\left[f(x) \neq \operatorname{sign}(h(x))\right] = \frac{1}{2^n} \sum_x \mathbb{1}_{\{f(x) \neq \operatorname{sign}(h(x))\}}$$

$\updownarrow$ compare these term-by-term to get Thm.

Consider "$(f(x)-h(x))^2$" vs. "$\mathbb{1}_{\{f(x) \neq \operatorname{sign}(h(x))\}}$":

Case 1 if $f(x) = \operatorname{sign}(h(x))$:

$$\mathbb{1}_{f(x) \neq \operatorname{sign}(h(x))} = 0$$

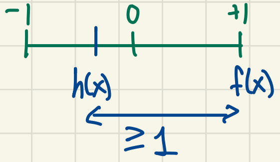
$$(f(x)-h(x))^2 \geq 0$$

Case 2 if $f(x) \neq \operatorname{sign}(h(x))$:

$$\mathbb{1}_{f(x) \neq \operatorname{sign}(h(x))} = 1$$

$$(f(x)-h(x))^2 \geq 1$$

why? e.g. if $f(x) = +1$ then in this case $h(x) < 0$:



(other case is analogous)

So, $\forall x$

$$(f(x)-h(x))^2 \geq \mathbb{1}_{f(x) \neq \operatorname{sign}(h(x))}$$ ∎

Correctness of learning algorithm

<u>Thm</u> if $\mathcal{C}$ has Fourier concentration $d = \alpha(\varepsilon, n)$

then there is a $q = O\left(\frac{n^d}{\varepsilon} \log \frac{n^d}{\delta}\right)$ sample uniform distribution learning algorithm for $\mathcal{C}$

i.e. algorithm gets q samples & with prob $\geq 1-\delta$ outputs h' s.t. $\Pr[f \neq h'] \leq 2 \cdot \varepsilon$

<u>Pf</u>.

run low degree alg with $\gamma = \varepsilon$

thm1 ⇒ get h s.t. $E[(f-h)^2] \leq \varepsilon + \varepsilon = 2\varepsilon$

output $h' := \text{sign}(h)$

$\Uparrow$

thm 2 ⇒ h' has error $\leq 2 \cdot \varepsilon$

∎

Applications

1) Bounded depth decision trees

$1\{x \text{ reaches leaf } \ell\}$

$$f(x) = \sum_{\substack{\ell \in \text{leaves} \\ \text{of } T}} \underbrace{f_\ell(x)}_{\substack{\text{fctn} \\ \text{which} \\ \text{depends on} \\ \le \text{depth many} \\ \text{vars}}} \cdot \underbrace{val(\ell)}_{\text{const}}$$

$$\hat{f}(S) = \sum val(\ell) \underbrace{\hat{f}_\ell(S)}_{\substack{0 \text{ for} \\ |S| > \text{depth}}}$$

linearity of Fourier representation

$$\Longrightarrow \quad \forall\, S \text{ s.t. } |S| > \text{depth}, \quad \hat{f}(S) = 0$$

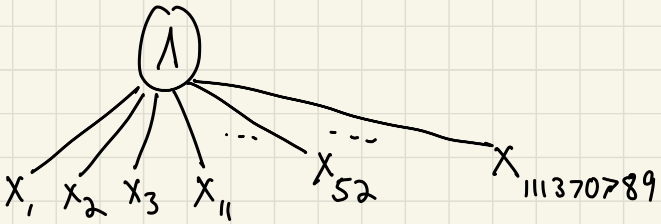
so $O\left(\frac{n^{\text{depth}}}{\varepsilon} \log \frac{n^{\text{depth}}}{\delta} \right)$ suffices

2) Constant depth ckts

def. "Boolean Ckt C" is DAG

gates: $\underbrace{\wedge, \vee, \neg}_{\text{operations}}, \underbrace{1, 0}_{\text{consts}}, \underbrace{x_1 \cdots x_n}_{\text{vars}}$

how many inputs? const, poly, unbounded?



Can we compute parity (xor) of n bits

in const depth?

yes! can compute any fctn on n bits in const depth "Karnaugh maps" parity in const depth, poly size ?

no! [Furst Saxe Sipser) ξ lemon

Switching lemma

lemons ⇒ lemonade:

<u>Thm</u> [Hastad, Linial Mansour Nisan]

$\forall f$ computable via size s depth d ckts

$$\sum_{|S|>t} \hat{f}^2(S) \le \alpha \quad \text{for} \quad t = O\left(\log \frac{s}{\alpha}\right)^{d-1}$$

$$\text{take} \quad \left.\begin{array}{l} s = \text{poly}(n) \\ d = \text{const} \\ \alpha = O(\varepsilon) \end{array}\right\} \Rightarrow t = O\left(\log^{d}\left(\tfrac{n}{\varepsilon}\right)\right)$$

yields $n^{O(\log^{d}(\frac{n}{\varepsilon}))}$ sample algorithm

(can improve to $n^{O(\log\log n)}$ [Jackson])

(recall parity of S will have 1 large Fourier coeff of degree $|S|$)

3) Learning halfspaces

$\text{sign}((\sum w_i x_i) - \theta)$

def. $h(x) = \text{sign}(w \cdot x - \theta)$ is "halfspace function"

$$\text{sign}(y) = \begin{cases} +1 & \text{if } y \geq 0 \\ -1 & \text{o.w.} \end{cases}$$

Thm Let h be halfspace over $\{\pm 1\}^n$ then h has f.c. $\alpha(\varepsilon) = \frac{c}{\varepsilon^2}$

$$\left(\text{i.e. } \sum_{|S| \geq \frac{c}{\varepsilon^2}} \hat{h}(S)^2 \leq \varepsilon\right)$$

(will prove soon)

Corr low degree alg learns halfspaces under unif dist with $n^{O(1/\varepsilon^2)}$ unif. samples.

(actually $O(n^5)$ sample algorithms exist, but this approach will have "big win" soon)

Useful idea to bound Fourier Concentration:

Noise Sensitivity

def. "Noise operator" $0 < \varepsilon < 1/2$

$$N_\varepsilon(x) = \text{randomly flip each bit of } x \text{ with prob } \varepsilon$$

def "Noise sensitivity"

$$NS_\varepsilon(f) = \Pr_{\substack{x \in \{\pm 1\}^n \\ \& \\ \text{noise}}} [f(x) \neq f(N_\varepsilon(x))]$$

Examples

1. $f(x) = x_1$ $\qquad ns_\varepsilon(f) = \varepsilon$

2. $f(x) = x_1 x_2 \ldots x_k$ $\qquad ns_\varepsilon(f) = \Pr[f(x) = F \;\&\; f(N_\varepsilon(x)) = T]$

$$+ \Pr[f(x) = T \;\&\; f(N_\varepsilon(x)) = F]$$

by symmetry → $= 2 \cdot \Pr[f(x) = T \;\&\; f(N_\varepsilon(x)) = F]$

if $\varepsilon << \frac{1}{k}$, $\approx \frac{1}{2^{k-1}}(\varepsilon k)$ ⟶ $= \frac{2}{2^k} \left(1 - (1-\varepsilon)^k\right)$ ⟵ $\Pr[\text{flip}]$

if $\varepsilon >> 1/k$, $\approx (1 - e^{-k\varepsilon}) / 2^{k-1}$

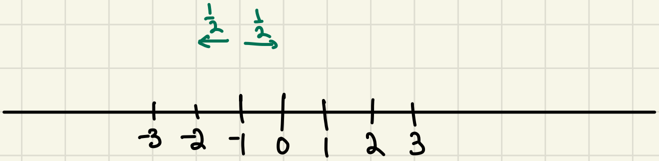
⟵ $\Pr[f(x) = T]$

3. $f(x) = \text{Maj}(x_1 \cdots x_n)$

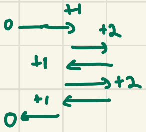
$$ns_\varepsilon(f) = O(\sqrt{\varepsilon})$$

high level sketch:

Maj(x) ~ random walk on line starting at 0



e.g. $x = (\underset{1}{1}\ \underset{2}{1}\ \underset{1}{-1}\ \underset{2}{1}\ \underset{1}{-1}\ \underset{0}{-1})$



well known fact:

$E[|x_1 + x_2 + \ldots + x_n|]$

$= \sqrt{n}$

+ likely to be close to $\sqrt{n}$

$N_\varepsilon(x)$ ~ random walk on $\underbrace{\varepsilon n}$ bits

each flip displaces by ±2 $(-1 \to +1$ or $+1 \to -1)$

$$E[\text{displacement}] = 2\sqrt{\varepsilon n}$$

Consider an "equivalent"

process: take walk specified by x + continue walk according to $N_\varepsilon(x) \cdot 2$

heuristic argument:

pretend first walk leaves us at $\sqrt{n}$

$\Pr[2^{nd}$ walk takes us back across $0]$

$$= \frac{1}{2} \Pr[2^{nd} \text{ displacement} > \underbrace{\sqrt{n}}_{\frac{1}{2\sqrt{\varepsilon}} \cdot \underbrace{2\sqrt{\varepsilon n}}_{\text{expected displacement}}}]$$

$\leq 2\sqrt{\varepsilon}$ by Markov's ≠

4. any LTF (½ space)

Thm (Peres) $NS_\varepsilon(LTF) < 8.8\sqrt{\varepsilon}$

best possible since $NS_\varepsilon(Maj) = \Theta(\sqrt{\varepsilon})$

5. Parity fctns $\chi_S(x)$ for $|S| = k$

$ns_\varepsilon(f) = \Pr[$odd # bits in S flipped by $N_\varepsilon]$

$$= \frac{1-(1-2\varepsilon)^k}{2}$$

for $|S|=1$: ε

6. Any f

Thm $f: \{\pm 1\}^n \to \{\pm 1\}$

$$NS_\varepsilon(f) = \frac{1}{2} - \frac{1}{2} \sum_S (1-2\varepsilon)^{|S|} \hat{f}(S)^2$$

for parity fcfns: $\frac{1}{2} - \frac{1}{2}(1-2\varepsilon)^{|S|}$

Pf. homework

Noise Sensitivity vs. Fourier Concentration

Thm $\forall f : \{\pm 1\}^n \to \{\pm 1\} \qquad 0 < \gamma < \frac{1}{2}$

$$\sum_{|S| \ge \frac{1}{\gamma}} \hat{f}(S)^2 < 2.32 \, ns_\gamma(f)$$

Pf

$$\begin{aligned}
2 \cdot ns_\gamma(f) &= 1 - \sum_S (1-2\gamma)^{|S|} \hat{f}(S)^2 && \text{previous thm} \\
&= \sum_S \hat{f}(S)^2 - \sum_S (1-2\gamma)^{|S|} \hat{f}(S)^2 && \text{Boolean Parseval} \\
&= \sum_S \left[1 - (1-2\gamma)^{|S|}\right] \hat{f}(S)^2 \\
&\ge \sum_{\substack{S \text{ s.t.} \\ |S| \ge 1/\gamma}} \left[1 - (1-2\gamma)^{1/\gamma}\right] \hat{f}(S)^2 \\
&> \sum_{|S| \ge 1/\gamma} (1 - e^{-2}) \hat{f}(S)^2
\end{aligned}$$

So $$\sum_{|S| \ge 1/\gamma} \hat{f}(S)^2 < \underbrace{\left(\frac{2}{1-e^{-2}}\right)}_{2.32} \cdot ns_\gamma(f)$$

∎

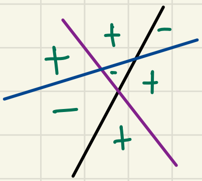
Corr for halfspace $h : \{\pm 1\}^n \to \{\pm 1\}$

$$\sum_{|S| \geq O(\frac{1}{\varepsilon^2})} \hat{f}(S) \leq \varepsilon$$

(pf omitted - some calculations + bound on NS)

$\Longrightarrow$ Can learn any halfspace from $n^{O(1/\varepsilon^2)}$ random examples

(actually can do a lot better)



Corr any function of k halfspaces can be learned with $n^{O(k^2/\varepsilon^2)}$ samples

e.g. parity of k vars, $\wedge$ of ½spaces

Pf idea noise sensitivity $\leq 8.8 k \sqrt{\varepsilon}$ by union bound.